

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2001-027986

(43)Date of publication of application : 30.01.2001

(51)Int.Cl.

G06F 13/14

G06T 1/20

H04N 1/41

(21)Application number : 2000-103492

(71)Applicant : CANON INC

(22)Date of filing : 05.04.2000

(72)Inventor : YAGUCHI HIROYUKI

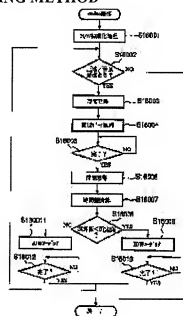
(30)Priority

Priority number : 11128907 Priority date : 10.05.1999 Priority country : JP

(54) DATA PROCESSOR AND PROCESSING PART SELECTING METHOD

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a data processor which can process data supplied from outside at high speed.
SOLUTION: The data processor having a software process part which actualizes a specific data process through software and a hardware process part which actualizes a specific data process through hardware acquires the process time of the hardware process part by using dummy data at a data process request (steps S16003 to S16005), and performs the process by the hardware when the time needed for the process is shorter than a specific time (step S16009) or by the software when longer (step S16011).



(51) Int.Cl. ⁷	識別記号	F I	テーマコード [*] (参考)
G 0 6 F 13/14	3 3 0	G 0 6 F 13/14	3 3 0 E
G 0 6 T 1/20		G 0 6 T 1/20	A
H 0 4 N 1/41		H 0 4 N 1/41	Z

審査請求 未請求 請求項の数33 O L (全 27 頁)

(21) 出願番号 特願2000-103492(P2000-103492)

(22) 出願日 平成12年4月5日 (2000. 4. 5)

(31) 優先権主張番号 特願平11-128907

(32) 優先日 平成11年5月10日 (1999. 5. 10)

(33) 優先権主張国 日本 (J P)

(71) 出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72) 発明者

矢口 博之

東京都大田区下丸子3丁目30番2号 キヤ

ノン株式会社内

(74) 代理人 100076428

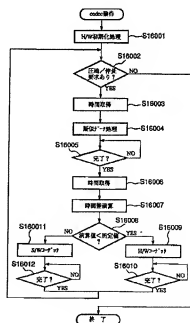
弁理士 大塚 康徳 (外2名)

(54) 【発明の名称】 データ処理装置及び処理部選択方法

(57) 【要約】

【課題】 外部から供給されるデータを高速に処理することのできるデータ処理装置を提供すること。

【解決手段】 所定のデータ処理をソフトウェアによって実現するソフトウェア処理部と、所定のデータ処理をハードウェアによって実現するハードウェア処理部とを有するデータ処理装置において、データ処理要求があった場合に疑似データを用いてハードウェア処理部の処理時間を取得し (ステップ S16003～S16005)、処理に要した時間が所定時間未満である場合にはハードウェアで (ステップ S16009)、所定時間以上である場合にはソフトウェアで処理を行う (ステップ S16011)。



【特許請求の範囲】

【請求項 1】 所定のデータ処理をソフトウェアによって実現するソフトウェア処理部と、前記所定のデータ処理をハードウェアによって実現するハードウェア処理部とを有し、前記ハードウェア処理部が前記所定のデータ処理に使用するバス手段が他の処理手段にも使用可能なデータ処理装置において、

テスト用データを用いて前記ハードウェア処理部に前記所定のデータ処理の実行を指示する手段と、前記テスト用データの処理に要した時間に応じてその後行う前記所定のデータ処理に使用する処理部を選択する選択手段とを有することを特徴とするデータ処理装置。

【請求項 2】 前記選択手段が、前記テスト用データの処理に要した時間が所定時間以上の場合には前記ソフトウェア処理部と、それ以外の場合には前記ハードウェア処理部を選択することを特徴とする請求項 1 記載のデータ処理装置。

【請求項 3】 前記所定のデータ処理を施すデータが発生するデータ発生手段が前記バス手段に接続されていることを特徴とする請求項 2 記載のデータ処理装置。

【請求項 4】 前記所定のデータ処理の結果を用いるデータ利用手段が前記バス手段に接続されていることを特徴とする請求項 2 記載のデータ処理装置。

【請求項 5】 前記所定のデータ処理を施すデータが発生するデータ発生手段及び前記所定のデータ処理の結果を用いるデータ利用手段がいずれも前記バス手段に接続されていることを特徴とする請求項 2 記載のデータ処理装置。

【請求項 6】 前記データ発生手段が画像読み取り装置であることを特徴とする請求項 3 又は請求項 5 のいずれかに記載のデータ処理装置。

【請求項 7】 前記データ利用手段が画像形成装置であることを特徴とする請求項 4 又は請求項 5 のいずれかに記載のデータ処理装置。

【請求項 8】 前記データ処理が符号化あるいは復号化処理であることを特徴とする請求項 1 乃至請求項 5 のいずれかに記載のデータ処理装置。

【請求項 9】 所定のデータ処理をソフトウェアによって実現するソフトウェア処理部と、前記所定のデータ処理をハードウェアによって実現するハードウェア処理部とを有し、前記ハードウェア処理部が前記所定のデータ処理に使用するバス手段が他の処理手段にも使用可能なデータ処理装置における処理部選択方法であって、テスト用データを用いて前記ハードウェア処理部に前記所定のデータ処理の実行を指示するステップと、前記テスト用データの処理に要した時間に応じてその後行う前記所定のデータ処理に使用する処理部を選択する選択ステップとを有することを特徴とする処理部選択方法。

【請求項 10】 前記選択ステップが、前記テスト用デ

ータの処理に要した時間を測定する測定ステップと、前記測定ステップで測定した時間が所定時間以上の場合には前記ソフトウェア処理部と、それ以外の場合には前記ハードウェア処理部を選択する判定ステップとを有することを特徴とする請求項 9 記載の処理部選択方法。

【請求項 11】 前記データ処理が符号化あるいは復号化処理であることを特徴とする請求項 9 又は請求項 10 に記載の処理部選択方法。

【請求項 12】 所定のデータ処理をソフトウェアによって実現するソフトウェア処理部と、前記所定のデータ処理をハードウェアによって実現するハードウェア処理部とを有し、前記ハードウェア処理部が前記所定のデータ処理に使用するバス手段が他の処理手段にも使用可能なデータ処理装置における処理部選択プログラムを格納したコンピュータ装置読み取り可能な記憶媒体であって、

テスト用データを用いて前記ハードウェア処理部に前記所定のデータ処理の実行を指示する工程のプログラムと、

前記テスト用データの処理に要した時間に応じてその後行う前記所定のデータ処理に使用する処理部を選択する選択工程のプログラムとを有することを特徴とする記憶媒体。

【請求項 13】 前記選択工程のプログラムが、前記テスト用データの処理に要した時間を測定する測定工程のプログラムと、

前記測定工程のプログラムで測定した時間が所定時間以上の場合には前記ソフトウェア処理部と、それ以外の場合には前記ハードウェア処理部を選択する判定工程のプログラムとを有することを特徴とする請求項 12 記載の記憶媒体。

【請求項 14】 前記所定のデータ処理が符号化あるいは復号化処理であることを特徴とする請求項 12 又は請求項 13 に記載の記憶媒体。

【請求項 15】 所定のデータ処理をソフトウェアによって実現するソフトウェア処理部と、前記所定のデータ処理をハードウェアによって実現するハードウェア処理部と、

前記所定の処理に対する要求を検出する要求検出手段と、

データ処理装置の動作状態に基づいて、前記要求のあった前記所定の処理を行う処理部を選択する選択手段とを有することを特徴とするデータ処理装置。

【請求項 16】 前記ソフトウェア処理部及び前記ハードウェア処理部の利用状況に基づいて前記選択手段が前記処理部の選択を行うことを特徴とする請求項 15 記載のデータ処理装置。

【請求項 17】 前記利用状況が、使用中であるか否かであることを特徴とする請求項 16 記載のデータ処理装置。

【請求項 18】 前記選択手段が、前記要求に対して利用可能な処理部が存在しない場合に、前記要求を記憶する記憶手段を更に有し、処理部が利用可能になった時点で前記記憶手段に記憶された要求があった場合、前記利用可能になった処理部を前記記憶された要求に基づく処理に選択することを特徴とする請求項 16 記載のデータ処理装置。

【請求項 19】 前記選択手段が、前記ハードウェア処理部及び前記ソフトウェア処理部の両方が利用可能である場合、前記ハードウェア処理部を優先的に選択することを特徴とする請求項 16乃至請求項 18のいずれか 1 項に記載のデータ処理装置。

【請求項 20】 前記ソフトウェア処理部及び/又は前記ハードウェア処理部を複数有することを特徴とする請求項 15乃至請求項 19のいずれか 1 項に記載のデータ処理装置。

【請求項 21】 前記データ処理が符号化あるいは復号化処理であることを特徴とする請求項 15乃至請求項 18のいずれか 1 項に記載のデータ処理装置。

【請求項 22】 所定のデータ処理をソフトウェアによって実現するソフトウェア処理部と、前記所定のデータ処理をハードウェアによって実現するハードウェア処理部とを有するデータ処理装置における、処理部選択方法であって、
前記所定の処理に対する要求を検出する要求検出ステップと、

所定の条件に基づいて、前記要求のあった前記所定の処理を行う処理部を選択する選択ステップとを有することを特徴とする処理部選択方法。

【請求項 23】 前記ソフトウェア処理部及び前記ハードウェア処理部の利用状況に基づいて、前記選択ステップが前記処理部の選択を行うことを特徴とする請求項 22 記載の処理部選択方法。

【請求項 24】 前記利用状況が、使用中であるか否かであることを特徴とする請求項 23 記載の処理部選択方法。

【請求項 25】 前記選択ステップが、前記要求に対して利用可能な処理部が存在しない場合に、前記要求を記憶する記憶ステップを更に有し、処理部が利用可能になった時点で前記記憶ステップに記憶された要求があった場合、前記利用可能になった処理部を前記記憶された要求に基づく処理に選択することを特徴とする請求項 23 記載の処理部選択方法。

【請求項 26】 前記選択ステップが、前記ハードウェア処理部及び前記ソフトウェア処理部の両方が利用可能である場合、前記ハードウェア処理部を優先的に選択することを特徴とする請求項 22乃至請求項 25のいずれか 1 項に記載の処理部選択方法。

【請求項 27】 前記データ処理装置が、前記ソフトウェア処理部及び/又は前記ハードウェア処理部を複数有

することを特徴とする請求項 22乃至請求項 26のいずれか 1 項に記載の処理部選択方法。

【請求項 28】 前記データ処理が符号化あるいは復号化処理であることを特徴とする請求項 22乃至請求項 25のいずれか 1 項に記載の処理部選択方法。

【請求項 29】 所定のデータ処理をソフトウェアによって実現するソフトウェア処理部と、前記所定のデータ処理をハードウェアによって実現するハードウェア処理部とを有するデータ処理装置における、処理部選択プログラムを格納したコンピュータ装置組み取り可能な記憶媒体であって、
前記所定の処理に対する要求を検出する要求検出工程のプログラムと、
前記ソフトウェア処理部及び前記ハードウェア処理部の利用状況に基づいて、前記要求のあった前記所定の処理を行う処理部を選択する選択工程のプログラムとを有することを特徴とする記憶媒体。

【請求項 30】 前記利用状況が、使用中であるか否かであることを特徴とする請求項 29 記載の記憶媒体。

【請求項 31】 前記選択工程のプログラムが、前記要求に対して利用可能な処理部が存在しない場合に、前記要求を記憶する記憶工程のプログラムを更に有し、処理部が利用可能になった時点で前記記憶工程のプログラムに記憶された要求があった場合、前記利用可能になった処理部を前記記憶された要求に基づく処理に選択することを特徴とする請求項 29 記載の記憶媒体。

【請求項 32】 前記選択工程のプログラムが、前記ハードウェア処理部及び前記ソフトウェア処理部の両方が利用可能である場合、前記ハードウェア処理部を優先的に選択することを特徴とする請求項 29乃至請求項 31のいずれか 1 項に記載の記憶媒体。

【請求項 33】 前記データ処理装置が、前記ソフトウェア処理部及び/又は前記ハードウェア処理部を複数有することを特徴とする請求項 29乃至請求項 32のいずれか 1 項に記載の記憶媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は画像データ等の入力データに所定の処理を施して出力するデータ処理装置に関し、特に複数の処理を高速に行うことの出来るデータ処理装置に関する。

【0002】

【従来の技術】ネットワーク技術の普及に伴い、ネットワーク上で画像、音声、プログラム等のデータのやり取りが頻繁に行われるようになってきた。多くの場合、ネットワーク負荷を減らし、高速にデータ伝送を行うためデータ圧縮（符号化）技術が用いられている。符号化・復号化は符号化復号化器（コーデック）を用いて行うが、コーデックは専用 LSI などのハードウェア（ハードウェアコーデック）によっても、汎用プロセッサによ

りソフトウェアを実行すること（ソフトウェアコーデック）によっても実現できる。一般には高速な圧縮を行う必要性があればハードウェアコーデック、処理速度がそれほど重要でない場合やコストを安価に抑える必要がある場合にはソフトウェアコーデックというように、ハードウェアコーデックあるいはソフトウェアコーデックのどちらか一方を用いている。

【0003】

【発明が解決しようとする課題】しかし、ネットワーク、スキャナ、プリンタへのインタフェースだけでなくハードウェアコーデックをPCIバスなどの共通バス上に接続する構成では、共通バスを使用することでインターフェースの統一による流用、実装のしやすさ、低コスト化などの利点がある反面、バスの同時使用者が多いとバスの競合が生じ、ハードウェアコーデックであるにもかかわらず、処理に時間がかかるという問題がある。例えばスキャナ、プリンタ、ハードウェアコーデックが共通バスに接続された構成においては、スキャナを用いて読みとった原稿の画像データを符号化してネットワーク上に接続された他の機器へ出力する処理と、他の機器から受信した画像データを復号化してプリンタ機能を用いて出力する処理が重複して発生したような場合においては、バス使用要求が重複し、ハードウェアコーデックを用いても処理に時間がかかることになる。

【0004】一方、最近では汎用プロセッサ（CPU）の性能も向上し、ソフトウェアコーデックでもある程度高速な処理が実現できるようになってきた。

【0005】本発明の目的は、外部から供給されるデータを高速に処理することのできるデータ処理装置を提供することにある。

【0006】また本発明の別の目的は、所定のデータ処理をソフトウェアによって実現するソフトウェア処理部と、所定のデータ処理をハードウェアによって実現するハードウェア処理部とを有するデータ処理装置において、処理時点で最適な処理部を選択可能な処理部選択方法を提供することにある。

【0007】

【課題を解決するための手段】すなわち、本発明の要旨は、所定のデータ処理をソフトウェアによって実現するソフトウェア処理部と、所定のデータ処理をハードウェアによって実現するハードウェア処理部とを有し、ハードウェア処理部が所定のデータ処理に使用するバス手段が他の処理手段にも使用可能なデータ処理装置において、テスト用データを用いてハードウェア処理部に所定のデータ処理の実行を指示する手段と、テスト用データの処理に要した時間に応じてその後行う所定のデータ処理に使用する処理部を選択する選択手段とを有することを特徴とするデータ処理装置に存する。

【0008】また、本発明の別の要旨は、所定のデータ処理をソフトウェアによって実現するソフトウェア処理部と、

所定のデータ処理をハードウェアによって実現するハードウェア処理部とを有し、ハードウェア処理部が所定のデータ処理に使用するバス手段が他の処理手段にも使用可能なデータ処理装置における処理部選択方法であって、テスト用データを用いてハードウェア処理部に所定のデータ処理の実行を指示するステップと、テスト用データの処理に要した時間に応じてその後行う所定のデータ処理に使用する処理部を選択する選択ステップとを有することを特徴とする処理部選択方法に存する。

【0009】また、本発明の別の要旨は、所定のデータ処理をソフトウェアによって実現するソフトウェア処理部と、所定のデータ処理をハードウェアによって実現するハードウェア処理部とを有し、ハードウェア処理部が所定のデータ処理に使用するバス手段が他の処理手段にも使用可能なデータ処理装置における処理部選択プログラムを格納したコンピュータ装置読込取り可能な記憶媒体であって、テスト用データを用いてハードウェア処理部に所定のデータ処理の実行を指示する工程のプログラムと、テスト用データの処理に要した時間に応じてその後行う所定のデータ処理に使用する処理部を選択する選択工程のプログラムとを有することを特徴とする記憶媒体に存する。

【0010】また、本発明の別の要旨は、所定のデータ処理をソフトウェアによって実現するソフトウェア処理部と、所定のデータ処理をハードウェアによって実現するハードウェア処理部と、所定の処理に対する要求を検出する要求検出手段と、データ処理装置の動作状態に基づいて、要求のあった所定の処理を行う処理部を選択する選択手段とを有することを特徴とするデータ処理装置に存する。

【0011】また、本発明の別の要旨は、所定のデータ処理をソフトウェアによって実現するソフトウェア処理部と、所定のデータ処理をハードウェアによって実現するハードウェア処理部とを有するデータ処理装置における、処理部選択方法であって、所定の処理に対する要求を検出する要求検出ステップと、所定の条件に基づいて、要求のあった所定の処理を行う処理部を選択する選択ステップとを有することを特徴とする処理部選択方法に存する。

【0012】また、本発明の別の要旨は、所定のデータ処理をソフトウェアによって実現するソフトウェア処理部と、所定のデータ処理をハードウェアによって実現するハードウェア処理部とを有するデータ処理装置における、処理部選択プログラムを格納したコンピュータ装置読み取り可能な記憶媒体であって、所定の処理に対する要求を検出する要求検出工程のプログラムと、ソフトウェア処理部及びハードウェア処理部の利用状況に基づいて、要求のあった所定の処理を行う処理部を選択する選択工程のプログラムとを有することを特徴とする記憶媒体に存する。

【0013】

【発明の実施の形態】（第1の実施形態）以下、図面を参照して本発明の好適な実施形態について詳細に説明する。以下の実施形態においては、本発明の情報処理装置の一実施形態である画像データの符号化復号化装置をネットワーク接続可能な複合機に適用した例を示す。

【0014】【ハードウェア】複合機システムの全体構成を図1に示す。コントローラ2000は、画像入力デバイスであるスキャナ2070や画像出力デバイスであるプリンタ2095と接続し、一方ではLAN2011や公衆回線（WAN）2051に接続することで、画像情報やデバイス情報の入出力を行うためのコントローラである。CPU2001は複合機全体を制御する汎用プロセッサである。RAM2002はCPU2001が動作するためのシステムワークメモリであり、画像データを一時的記憶するための画像メモリでもある。

【0015】ROM2003はブートROMであり、複合機のブートプログラムが格納されている。HDD2004はハードディスクドライブで、システムソフトウェア、画像データを格納する。操作部1/F2006は操作部（U1）2012とのインターフェース部で、操作部2012に表示する画像データを操作部2012に対して出力する。また、操作部2012から本システム使用者が入力した情報を、CPU2001に伝える役割をする。Modem2050は公衆回線2051に接続し、情報の入出力を行う。以上のデバイスがシステムバス2007上に配置される。

【0016】画像バスI/F2005はシステムバス2007と画像データを高速で伝送する画像バス2008を接続し、データ構造を変換するバスブリッジである。画像バス2008は、PCIバスまたはIEEE1394で構成される。画像バス2008上には以下のデバイスが配置される。ネットワーク1/F2010は画像バス2008をLAN2011に接続し、情報の入出力を行う。ラスターイメージプロセッサ（RIP）2060はPDL（ページ記述言語）コードをビットマップイメージに展開する。

【0017】デバイスI/F部2020は、画像入出力デバイスであるスキャナ2070やプリンタ2095とコントローラ2000を接続し、画像データの同期系/非同期系の変換を行う。スキャナ画像処理部2080は、入力画像データに対し補正、加工、編集を行う。プリンタ画像処理部は、プリント出力画像データに対して、プリンタの補正、解像度変換等を行う。画像回転部2030は画像データの回転を行う。コーデック2040は、多値画像データはJPEG、2値画像画像データはJBIG、MMR、MHの圧縮伸張処理を行う。

【0018】以上説明したような構成は画像処理部分の拡張性を考慮してシステムバス2007部、及び画像バス2008部が分離されるような構成となっており、一

般的なコンピュータの構成を応用したものである。上記構成では画像バスI/Fを汎用I/Fにすることで、画像処理を任意に組み合わせることが可能な自由度、また将来性を考慮し拡張性をもたせている。特にコーデック部分は将来様々な規格が提案される可能性もあり、容易に交換できるように画像バス側に接続される。

【0019】複合機システムの正面外観例を図2に示す。画像入力デバイスであるスキャナ部2070は、原稿を照明し、CCDラインセンサ（図示せず）を走査することで、ラスターイメージデータとして電気信号に変換する。原稿は原稿フィード2072のトレイ2073にセットし、装置使用者が操作部2012から読み取り起動指示することにより、コントローラCPU2001がスキャナ2070に指示を与え、フィード2072は原稿を1枚ずつフィードし原稿画像の読み取り動作を行う。

【0020】画像出力デバイスであるプリンタ部2095は、ラスターイメージデータ2096を紙などの被記録材上の画像に変換する部分であり、その方式は感光体ドラムや感光体ベルトを用いた電子写真方式、微小ノズルアレイからインクを吐出して用紙上に直接画像を形成するインクジェット方式等があるが、どの方式でも構わない。プリント動作の起動は、コントローラCPU2001からの指示によって開始する。プリンタ部2095には、異なる被記録材サイズまたは異なる被記録材の向きを選択できるように複数の供給段を持ち、それに対応したカセット2101、2102、2103、2104がある。また、排出トレイ2111は印字し終った被記録材を受けるものである。

【0021】・操作部
操作部2012の外観例を図3に示す。LCD表示部2013は、LCD上にタッチパネルシートが貼られており、システムの操作画面を表示するとともに、表示しているキーが押されるとその位置情報をコントローラCPU2001に伝える。スタートキー2014は原稿画像の読み取り動作を開始する時などに用いる。スタートキー2014中央部には、緑と赤の2色LED2018があり、その色によってスタートキー2014が使える状態にあるかどうかを示す。ストップキー2015は稼働中の動作を止める働きをする。IDキー2016は、使用者のユーザIDを入力する時に用いる。リセットキー2017は操作部からの設定を初期化する時に用いる。

【0022】・スキャナ画像処理部
スキャナ画像処理部2080の構成を図4に示す。画像バスI/Fコントローラ2081は、画像バス2008と接続し、そのバスアクセスシーケンスを制御する働きと、スキャナ画像処理部2080内の各デバイスの制御及びタイミングを発生させる。フルタ処理部2082は、空間フィルタでコンボリューション演算を行う。編

集部2083は、例えば入力画像データからマーカーペンで囲まれた閉領域を認識して、その閉領域内の画像データに対して、膨つけ、網掛け、ネガボ反転等の画像加工処理を行う。

【0023】変倍処理部2084は、読み取り画像の解像度を変える場合にラスターイメージの主走査方向について補間演算を行い拡大、縮小を行う。副走査方向の変倍については、画像読み取りラインセンサ（図示せず）を走査する速度を変えることで行う。テーブル変換部2085は、読み取った輝度データである画像データを濃度データに変換するのテーブル変換を行う。2値化処理部2086は、多値のグレースケール画像データを、誤差拡散処理やスクリーン処理によって2値化する。

【0024】処理が終了した画像データは、再び画像バスコントローラ2018を介して、画像バス上に転送される。

【0025】・プリンタ画像処理部

プリンタ画像処理部2090の構成を図5に示す。画像バス1/Fコントローラ2091は、画像バス2008と接続し、そのバスアクセスシーケンスを制御する働きと、スキャン画像処理部2090内の各デバイスの制御及びタイミングを発生させる。解像度変換部2092は、LAN2011あるいはWAN2051から受信した画像データを、プリンタ2095の解像度に変換するための解像度変換を行う。スムージング処理部2093は、解像度変換後の画像データのジャギー（斜め線等の白黒境界部に現れる画像のがつき）を滑らかにする処理を行う。

【0026】・コーデック

コーデック2040の構成を図6に示す。画像バス1/Fコントローラ2041は、画像バス2008と接続し、そのバスアクセスシーケンスを制御する働き、入力バッファ2042・出力バッファ2045とデータのやりとりを行うためのタイミング制御及び、画像圧縮伸長部2043に対するモード設定などの制御を行う。本発明においては、データの圧縮を符号化、伸長を復号化として取り扱う。以下にコーデックの処理手順を示す。

【0027】画像バス2008を介して、CPU2001から画像バス1/Fコントローラ2041に画像圧縮制御のための設定を行う。この設定により画像バス1/Fコントローラ2041は画像圧縮伸長部2043に対して画像圧縮に必要な設定（たとえばMMR圧縮・JBIG伸長等）を行う。必要な設定を行った後に、再度CPU2001から画像バス1/Fコントローラ2041に対して画像データ転送の許可を行う。この許可に従い、画像バス1/Fコントローラ2041はRAM2002もしくは画像バス2008上の各デバイスから画像データの転送を開始する。

【0028】受け取った画像データは入力バッファ2042に一時格納され、画像圧縮伸長部2043の画像デ

ータ要求に応じて一定のスピードで画像を転送する。この際、入力バッファは画像バス1/Fコントローラ2041と、画像圧縮伸長部2043両者の間で、画像データを転送できるかどうかを判断し、画像バス2008からの画像データの読み込み及び、画像圧縮伸長部2043への画像の書き込みが不可能である場合は、データの転送を行わないような制御を行う（以後このような制御をハンドシェイクと呼称する）。

【0029】画像圧縮伸長部2043は受け取った画像データを、一旦RAM2044に格納する。これは画像圧縮を行う際には行う画像圧縮処理の種類によって、数ライン分のデータを要するためであり、最初の1ライン分の圧縮を行うためには数ライン分の画像データを用意してからでないとい画像圧縮が行えないためである。画像圧縮を施された画像データは直ちに出力バッファ2045に送られる。

【0030】出力バッファ2045では、画像バス1/Fコントローラ2041及び画像圧縮伸長部2043とのハンドシェイクを行い、画像データを画像バス1/Fコントローラ2041に転送する。画像バス1/Fコントローラ2041では転送された圧縮（もしくは伸長）された画像データをRAM2002もしくは画像バス2008上の各デバイスにデータを転送する。こうした一連の処理は、CPU2001からの処理要求が無くなるまで（必要なべき数の処理が終わったとき）、もしくはこの画像圧縮伸長部から停止要求が出るまで（圧縮及び伸長時のエラー発生時等）繰り返される。

【0031】・画像回転部

画像回転部2030の構成を図7に示す。画像バス1/Fコントローラ2031は、画像バス2008と接続し、そのバスシーケンスを制御する働き、画像回転部2032にモード等を設定する制御及び、画像回転部2032に画像データを転送するためのタイミング制御を行う。以下に画像回転部の処理手順を示す。

【0032】画像バス2008を介して、CPU2001から画像バス1/Fコントローラ2031に画像回転制御のための設定を行う。この設定により画像バス1/Fコントローラ2031は画像回転部2032に対して画像回転に必要な設定（たとえば画像サイズや回転方向・角度等）を行う。必要な設定を行った後に、再度CPU2001から画像バス1/Fコントローラ2031に対して画像データ転送の許可を行う。この許可に従い、画像バス1/Fコントローラ2031はRAM2002もしくは画像バス2008上のデバイスから画像データの転送を開始する。尚、ここでは画像バス2008の幅を32bitとし回転を行う画像サイズを32×32（bit）とし、又、画像バス2008上に画像データを転送させる際に32bitを単位とする画像転送を行うものとする（扱う画像は2値を想定する）。

【0033】上述のように、32×32（bit）の画

像を得るためには、上述の単位データ転送を32回行う必要があり、且つ不連続なアドレスから画像データを転送する必要がある（図8参照）。不連続アドレスリングにより転送された画像データは、読み出し時に所望の角度に回転されているように、RAM2033に書き込まれる。例えば、90度反時計方向回転であれば、最初に転送された32ビットの画像データを、図9のようにY方向に書き込んでいく。読み出し時にX方向に読み出すことで、画像が回転される。

【0034】32×32（bit）の画像回転（RAM2033への書き込み）が完了した後、画像回転部2032はRAM2033から上述した読み出し方法で画像データを読み出し、画像バス1/Fコントローラ2031に画像を転送する。

【0035】回転処理された画像データを受け取った画像バス1/Fコントローラ2031は、連続アドレスリングを以て、RAM2002もしくは画像バス2008上の各デバイスにデータを転送する。こうした一連の処理は、CPU2001からの処理要求が無くなるまで（必要なページ数の処理が終わったとき）繰り返される。

【0036】・デバイス1/F部
デバイス1/F部2020の構成を図10に示す。画像バス1/Fコントローラ2021は、画像バス2008と接続し、そのバスアクセスシーケンスを制御する働きと、デバイス1/F部2020内の各デバイスの制御及びタイミングを発生させる。また、外部のスキナ2070及びプリンタ2095への制御信号を発生させる。スキャンバッファ2022は、スキナ2070から送られてくる画像データを一時保存し、画像バス2008に同期させて画像データを出力する。

【0037】シリアルパラレル・パラレルシリアル変換部2023は、スキャンバッファ2022に保存された画像データを順番に並べて、あるいは分解して、画像バス2008に転送できる画像データのデータ幅に変換する。パラレルシリアル・シリアルパラレル変換部2024は、画像バス2008から転送された画像データを分解して、あるいは順番に並べて、プリントバッファ2025に保存できる画像データのデータ幅に変換する。プリントバッファ2025は、画像バス2008から送られてくる画像データを一時保存し、プリンタ2095に同期させて画像データを出力する。

【0038】画像スキャン時の処理手順を以下に示す。スキナ2070から送られてくる画像データをスキナ2070から送られてくるタイミング信号に同期させて、スキャンバッファ2022に保存する。そして、画像バス2008がPCIバスの場合には、バッファ内に画像データが32ビット以上入ったときに、画像データを先入れ先出しで32ビット分、バッファからシリアルパラレル・パラレルシリアル変換2023に送り、32

ビットの画像データに変換し、画像バス1/Fコントローラ2021を通して画像バス2008上に転送する。また、画像バス2008がIEEE1394の場合には、バッファ内の画像データを先入れ先出しで、バッファからシリアルパラレル・パラレルシリアル変換部2023に送り、シリアル画像データに変換し、画像バス1/Fコントローラ2021を通して画像バス2008上に転送する。

【0039】画像プリント時の処理手順を以下に示す。画像バス2008がPCIバスの場合には、画像バスから送られてくる32ビットの画像データを画像バス1/Fコントローラで受け取り、パラレルシリアル・シリアルパラレル変換部2024に送り、プリンタ2095の入力データビット数の画像データに分解し、プリントバッファ2025に保存する。また、画像バス2008がIEEE1394の場合には、画像バスから送られてくるシリアル画像データを画像バス1/Fコントローラで受け取り、パラレルシリアル・シリアルパラレル変換部2024に送り、プリンタ2095の入力データビット数の画像データに変換し、プリントバッファ2025に保存する。そして、プリンタ2095から送られてくるタイミング信号に同期させて、バッファ内の画像データを先入れ先出しで、プリンタ2095に送る。

【0040】・ネットワークシステム
本発明のコーデックを適用した複合機をネットワークに接続した場合のシステム構成例を図11に示す。1001は本発明のコーデックを適用した複合機で、後述するスキナとプリンタから構成され、スキナから読み込んだ画像をローカルエリアネットワーク1010（以下LAN）に流したり、LANから受信した画像をプリンタによりプリントアウトできる。また、スキナから読んだ画像を図示しないFAX送信手段により、PSTNまたはISDN（1030）に送信したり、PSTNまたはISDNから受信した画像をプリンタによりプリントアウトできる。1002は、データベースサーバで、複合機1001により読み込んだ2値画像及び多値画像をデータベースとして管理する。1003は、データベースサーバ1002のデータベースクライアントで、データベース1002に保存されている画像データを閲覧／検索等できる。

【0041】1004は、電子メールサーバで、複合機1001により読み取った画像を電子メールの添付として受け取ることができる。1005は、電子メールのクライアントで、電子メールサーバ1004の受け取ったメールを受信し閲覧したり、電子メールを送信したり、可能である。1006がHTML文書をLANに提供するWWWサーバで、複合機1001によりWWWサーバで提供されるHTML文書をプリントアウトできる。1011は、ルータでLAN1010をインターネット／イントラネット1012と連結する。インターネット／

イントラネットに、前述したデータベースサーバ（1002）、WWWサーバ（1006）、電子メールサーバ（1004）、複合機（1001）と同様の装置が、それぞれ1020、1021、1022、1023として連結している。一方、複合機1001は、PSTNまたはISDN（1030）を介して、FAX装置1031と送受信可能になっている。また、LAN上にプリンタ1040も連結されており、複合機1001により読み取った画像をプリントアウト可能なように構成されている。

【0042】【ソフトウェア】図12は、図1に示した複合機のソフトウェアのモジュール構成を示すブロック図である。1501はUI即ちユーザインターフェイスを司るものであり、オペレータが本複合機の各種操作・設定を行う際、機器との仲介を行うモジュールである。本モジュールは、オペレータの操作に従い、後述の各種モジュールに入力情報を転送し処理の依頼、或いはデータの設定等を行う。

【0043】1502はAddress-Book即ちデータの送付先、通信先等を管理するデータベースモジュールである。Address-Bookの内容は、UI1501からの操作によりデータの追加、削除、取得が行われ、オペレータの操作により後述の各モジュールにデータの送付・通信先情報を与えるものとして使用されるものである。

【0044】1503はWeb-Serverモジュールであり、図外のWebクライアントからの要求により、本複合機の管理情報を提供するために使用される。管理情報は、後述のControl-API1518を介して読み取られ、後述のHTTP1512、TCP/IP1516、Network-Driver1517を介してWebクライアントに通知される。

【0045】1504はUniversal-Send即ち、データの配信を司るモジュールであり、UI1501によりオペレータに指示されたデータを、同様に指示された通信（出力）先に配布するものである。また、オペレータにより、本機器のスキヤナ機能を使用し配布データの生成が指示された場合は、後述のControl-API1518を介して機器を動作させ、データの生成を行う。

【0046】1505はUniversal-Send1504内で出力先にプリンタが指定された際に実行されるモジュールである。1506はUniversal-Send1504内で通信先にE-mailアドレスが指定された際に実行されるモジュールである。1507はUniversal-Send1504内で出力先にデータベースが指定された際に実行されるモジュールである。1508はUniversal-Send1504内で出力先に本機器と同様の複合機が指定された際に実行されるモジュールである。

【0047】1509はRemote-Copy-Scanモジュールであり、本複合機のスキヤナ機能を使用し、ネットワーク等で接続された他の複合機を出力先とし、本複合機単

体で実現しているCopy機能と同等の処理を行うモジュールである。1510はRemote-Copy-Printモジュールであり、本複合機のプリンタ機能を使用し、ネットワーク等で接続された他の複合機を入力先とし、本複合機単体で実現しているCopy機能と同等の処理を行うモジュールである。

【0048】1511はWeb-Pull-Print即ちインターネットまたはイントラネット上の各種ホームページの情報を読み出し、印刷するモジュールである。1512は本複合機がHTTPにより通信する際に使用されるモジュールであり、後述のTCP/IP1516モジュールにより前述のWeb-Server1503、Web-Pull-Print1511モジュールに通信を提供するものである。1513は1prモジュールであり、後述のTCP/IP1516モジュールにより前述のUniversal-Send1504内のプリンタモジュール1505に通信を提供するものである。

【0049】1514はSMTPモジュールであり、後述のTCP/IP1516モジュールにより前述のUniversal-Send1504内のE-mailモジュール1506に通信を提供するものである。

【0050】1515はSLM即ちSalutation-Managerモジュールであり、後述のTCP/IP1516モジュールにより前述のUniversal-Send1504内のデータベースモジュール1507、DPモジュール1508、及びRemote-Copy-Scan1509モジュール、Remote-Copy-Print1510モジュールに通信を提供するものである。1516はTCP/IP通信モジュールであり、前述の各種モジュールに後述のNetwork-Driverによりネットワーク通信を提供するものである。1517はネットワークドライバであり、ネットワークに物理的に接続される部分を制御するものである。

【0051】1518はControl-APIであり、Universal-Send1504等の上流モジュールに対し、後述のJob-Manager1519等の下流モジュールとのインターフェイスを提供するものであり、上流、及び下流のモジュール間の依存関係を軽減しそれぞれの流用性を高めるものである。1519はJob-Managerであり、前述の各種モジュールよりControl-API1518を介して指示される処理を解釈し、後述の各モジュールに指示を与えるものである。また、本モジュールは、本複合機内で実行されるハード的な処理を一元管理するものである。

【0052】1520はCODEC-Managerであり、Job-Manager1519が指示する処理の中でデータの各種圧縮・伸長を管理・制御するものである。1521はFBE-Encoderであり、Job-Manager1519、Scan-Manager1524により実行されるスキヤナ処理により読み込まれたデータをFBEフォーマットにより圧縮するものである。

【0053】1522はJPEG-CODECであり、Job-Manager

r1519、Scan-Manager1524により実行されるスキャン処理、及びPrint-Manager1526により実行される印刷処理において、読み込まれたデータのJPEG圧縮及び印刷データのJPEG展開処理を行うものである。

【0054】1523はMMR-CODECであり、Job-Manager1519、Scan-Manager1524により実行されるスキャン処理、及びPrint-Manager1526により実行される印刷処理において、読み込まれたデータのMMR圧縮及び印刷データのMMR伸長処理を行うものである。

【0055】1524はScan-Managerであり、Job-Manager1519が指示するスキャン処理を管理・制御するものである。1525はスキャナドライバであり、Scan-Manager1524と本複合機が内部的に接続しているスキャナ部との通信を行うものである。1526はPrint-Managerであり、Job-Manager1519が指示する印刷処理を管理・制御するものである。

【0056】1527はプリンタドライバであり、Print-Manager1526と印刷部とのI/Fを提供するものである。1528はパラレルポートドライバであり、Web-Pull-Print1511がパラレルポートを介して図外の出力機器にデータを出力する際のI/Fを提供するものである。

【0057】・アプリケーション

以下、本発明の組み込みアプリケーションの実施の形態について図面を用いて説明する。図13は、図1に示す複合機における、画像データ配信に関する組み込みアプリケーションブロックを表す図である。

【0058】4050は、図3で説明した操作部のアプリケーションを示すブロックである。4100は、リモートコピーアプリケーションの送信側を示すブロックである。4150は、同報配信の送信側を示すブロックである。4200は、Web Pull Printモジュールを示すブロックである。

【0059】4250は、Web Serverモジュールを示すブロックである。4300は、リモートコピーの受信側（プリント側）を示すブロックである。4350は、同報配信で送信されてきたイメージを汎用のプリンタで受信・プリントするブロックである。4400は、リモートプリントの受信側（プリント側）を示すブロックである。

【0060】4450は、同報配信で送信されてきたイメージを公知のNotes（商標）Serverで受信・格納するブロックである。4500は、同報配信で送信されてきたイメージを2値のイメージを受信・格納するブロックである。4550は、同報配信で送信されてきたイメージを公知のMail Serverで受信・格納するブロックである。

【0061】4600は、同報配信で送信されてきたイメージを多値のイメージを受信・格納するブロックであ

る。4650は、情報コンテンツを含んだ、公知のWeb Serverである。4700は、本発明のWeb Serverなどにアクセスする公知のWeb Browserである。

【0062】以下、それぞれのブロックに照らし合わせながら、アプリケーション群の説明を詳細に行う。

・User Interfaceアプリケーション

ブロック4050に示したUser Interface（以下、UI）の詳細は、前記したとおりであるが、ここでは、4051のAddress Bookについて説明する。このAddress Bookは、本発明の機器内の不揮発性の記憶装置（不揮発性メモリやハードディスクなど）に保存されており、この中には、ネットワークに接続された機器の特徴が記載されている。例えば、以下に列挙するようなものが含まれている。

機器の正式名やエイリアス名

機器のネットワークアドレス

機器の処理可能なネットワークプロトコル

機器の処理可能なドキュメントフォーマット

機器の処理可能な圧縮タイプ

機器の処理可能なイメージ解像度

プリンタ機器の場合の給紙可能な紙サイズ、給紙段情報

サーバー（コンピュータ）機器の場合のドキュメントを格納可能なフォルダ名

【0063】以下に説明する各アプリケーションは、上記Address Book 4051に記載された情報により配信先の特徴を判別することが可能となる。また、このAddress Bookは、編集可能であると共に、ネットワーク内のサーバーコンピュータなどに保存されているものをダウンロードして使用する、または、直接参照することも可能である。

【0064】・リモートコピーアプリケーション

リモートコピーアプリケーション4041は、配信先に指定された機器の処理可能な解像度情報を前記Address Book 4051より判別し、それに従い、スキャナにより読みとった画像2値画像を公知のMMR圧縮を用いて圧縮し、それを公知のTIFF（Tagged Image File Format）化し、SLM4103を通して、ネットワーク上のプリンタ機器に送信する。SLM4103とは、詳細には説明しないが、公知のSalutation Manager（または、Smart Link Manager）と呼ばれる機器制御情報などを含んだネットワークプロトコルの一種である。

【0065】・同報配信アプリケーション

同報配信アプリケーション4150は、前記リモートコピーアプリケーションと違い、一度の画像走査で複数の配信宛先に画像を送信する事が可能である。また、配信先もプリンタのような画像出力機器にとどまらず、いわゆるサーバーコンピュータにも直接配信可能である。

【0066】以下、配信先に従って順に説明する。配信

先の機器が公知のネットワークプリンタプロトコルである L P D (Line プリント Daemon)、プリンタ制御コマンドとして公知の L I P S を処理可能だと Address Book 4 0 5 1 より判別した場合、同様に Address Book 4 0 5 1 より判別した画像解像度に従って画像読み取りを行い、画像自体は、本実施形態では、公知の F B E (First Binari Encoding) を用いて圧縮し、さらに L I P S コード化して、公知のネットワークプリンタプロトコルである L P R で相手機器に送信する。

【0067】 配信先の機器が前記 S L M で通信可能で、サーバー機器の場合、Address Book 4 0 5 1 より、サーバーアドレス、サーバー内のフォルダの指定を判別し、リモートコピーアプリケーションと同様に、スキャナにより読みとった画像 2 値画像を公知の M M R 圧縮を用いて圧縮し、それを公知の T I F F (Tagged Image File Format) 化し、S L M を通して、ネットワーク上のサーバー機器の特定のフォルダに格納する事が可能である。

【0068】 また、本実施形態の機器では、相手機器であるサーバーが公知の J P E G 圧縮された多値画像を処理可能だと判別した場合、前記の 2 値画像と同様に多値読み取りした画像を公知の J P E G 圧縮を用いて、やはり公知の J F I F 化し、S L M を通して、ネットワーク上のサーバー機器の特定のフォルダに格納する事が可能である。

【0069】 配信先の機器が公知の E - M a i l サーバーである場合、Address Book 4 0 5 1 に記載されたメールアドレスを判別し、スキャナにより読みとった画像 2 値画像を公知の M M R 圧縮を用いて圧縮し、それを公知の T I F F (Tagged Image File Format) し、公知の S M T P (Simple Mail Transfer Protocol) 4 1 5 3 を使用して、E - M a i l サーバーに送信する。その後の配信は、Mail Server 4 5 5 0 に従って実行される。

【0070】 ・Web Pull Print アプリケーション
Web Pull Print アプリケーション 4 2 0 0 は、本実施形態と直接関係しないので、説明は省略する。

【0071】 ・Web Server アプリケーション
Web Server アプリケーションは 4 2 5 0、本実施形態と直接関係しないので、説明は省略する。

【0072】 以下、操作部の画面について簡単に説明する。

・操作画面

図 3 に示した操作部 2 0 1 2 の L C D 表示部 2 0 1 3 の表示例を図 1 4 に示す。図は、コピー機能の表示例を示している。

【0073】 本発明の装置が提供する機能は、Copy/Send/Retrieve/Tasks/Management/Configuration の 6 つの大きなカテゴリーに分かれており、これらは操作画面 (3 0 1 0) 上の上部に表示される 6 つのメインタブ (COPY/SEND/RETRIEVE/TASKS/MGMT/CONFIG) (3 0 1 1 ~ 3 0 1 6) に対応している。これらのメインタブを押すこ

とにより、各カテゴリーの画面への切り替えが行われる。他カテゴリーへの切り替えが許可されない場合は、メインタブの表示色が変わり、メインタブを押しても反応しない。

【0074】 Copy は自機が有するスキャナとプリンタを使用して通常のドキュメント複写を行う機能と、自機が有するスキャナとネットワークで接続されたプリンタを使用してドキュメントの複写を行う機能 (リモートコピー) を含む。Send は自機が有するスキャナに置かれたドキュメントを、電子メール、リモートプリンタ、ファックス、ファイル転送 (F T P) およびデータベースに転送する機能であり、宛先を複数指定することが可能である。

【0075】 Retrieve は外部にあるドキュメントを取得し、自機が有するプリンタで印刷する機能である。ドキュメントの取得手段として WWW、電子メール、ファイル転送およびファックスの使用が可能である。Tasks はファックスやインターネットプリントなどの外部から送られるドキュメントを自動処理し、定期的に Retrieve を行うためのタスクの生成、管理を行う。Management はジョブ・アドレス帳・ブックマーク・ドキュメント・アカウント情報などの管理を行う。Configuration では自機に関しての設定 (ネットワーク、時計など) を行う。

【0076】 ・Device Information Service (D I S)
コントロール内でジョブに対する設定値、デバイス (スキャナ、プリンタなど) の機能、ステータス、課金情報等を Control API に準拠したデータ形式で保持するデータベースと、そのデータベースとの I / F を Device Information Service (以下、D I S と呼称する) として定義している。図 1 5 に D I S 7 1 0 2 と Job Manager 1 5 1 9、及び Scan、Print の各 Manager 1 5 2 4、1 5 2 6 との関係を示す。

【0077】 基本的に、ジョブの開始命令など動的な情報は Job Manager 1 5 1 9 から各 Manager に直接指示され、デバイスの機能やジョブの内容など静的な情報は D I S 7 1 0 2 を参照する。各 Manager から動的な情報、イベントは D I S 7 1 0 2 を介して Job Manager 1 5 1 9 に伝えられる。

【0078】 各 Manager から D I S のデータベースにデータの設定、取得を行う場合、D I S の内部データ形式が Control API 準拠であることから、Control API に準拠したデータ形式と各 Manager が理解できるデータ形式との相互の変換処理を行う。例えば、各 Manager からステータスデータの設定を行う場合、デバイス固有のデータを解釈し、Control API で定義される対応するデータに変換し、D I S のデータベースへ書き込みを行う。

【0079】 Job Manager から D I S のデータベースにデータの設定、取得を行う場合には、Job Manager と D I S の間でデータの変換は生じない。

【0080】 また D I S には、Manager から通知される

各種イベント情報に基づき、イベントデータの更新が行われる。

【0081】図16にDIS内部に保持される各種データベース（以下、DBと呼称する）を示し、それぞれのDBについて説明する。

【0082】7201はSupervisor DBであり、機器全体についてのステータスやユーザ情報を保持しているDBであり、ユーザIDやパスワード等、バックアップが必要な情報はHD装置、あるいはバックアップメモリなどの不揮発性の記憶装置に保持される。

【0083】7202はScan Component DB、7203はPrint Component DBであり、これらComponent DBは存在するComponent毎に対応して保持される。例えば、プリンタのみからなる機器の場合はPrint Component DBのみが存在し、また例えば、FAXを備えた機器の場合はFAX Component DBが保持される。各Component DBには初期化時に、それぞれ対応するManagerがComponentの機能やステータスを設定する。

【0084】7204はScan Job Service DB、7205はPrint Job Service DBであり、これらのJob Service DBもComponent DB同様、初期化時にそれぞれ対応するManagerが機器で使用できる機能や、それらのサポート状況を設定する。

【0085】次にJob DB、Document DBについて説明する。7206はScan Job DB、7207はPrint Job DBの各Job DB、7208はScan Document DB、7209はPrint Document DBである。

【0086】Job DB、Document DBはJobとそれに付随するDocumentが生成される度にJobManagerにより動的に確保、初期化が行われ、必要な項目の設定が行われる。各ManagerはJobの処理開始前にJob DB、およびDocument DBから処理に必要な項目を読み出し、Jobを開始する。その後、Jobが終了するとこれらのJob、及びそれに付随していたDocumentのDBは解放される。Jobは1つ以上のDocumentを持つので、あるJobに対して複数のDocument DBが確保される場合がある。

【0087】7211は各Managerから通知されるイベント情報を保持するデータベース、7210は装置のScan回数、Print回数を記録するためのカウンタテーブルである。

【0088】Managerから通知されるイベントには、Scan ManagerからのComponentの状態遷移、Scan処理動作完了や各種のエラー、またPrint ManagerからのComponentの状態遷移、Print処理動作完了、紙詰まり、給紙カセットオープンなどがあり、それぞれのイベントを識別するためのイベントIDが予め定められている。

【0089】Managerからイベントが発行された場合、DISはイベントデータベース7211に発行されたイベントIDと必要ならイベントに付随する詳細データを

登録する。また、Managerからイベントの解除が通知された場合、解除指定されたイベントデータをイベントデータベース7211から削除する。

【0090】Job Managerよりイベントのポーリングが行われた場合、DISはイベントデータベース7211を参照し、現在発生しているイベントIDと必要ならイベントに付随する詳細データをJob Managerへ返信し、現在イベントが発生していなければその旨を返信する。

【0091】また、Scan処理動作完了、Print処理動作完了のイベントが通知された場合はScan、Printを行ったユーザのカウンタ値を更新する。このソフトウェアによるカウンタは不慮の電源遮断などでその値が失われるように、バックアップされたメモリ装置やHD装置の不揮発性記憶装置にその値が更新されるたびに書き戻す。

【0092】・スキャン動作

図17はスキャン動作に関するソフトウェア構造を示したものである。Job Manager1519はアプリケーションレベルの要求を分類、保存する機能を持つ。DIS7102はアプリケーションレベルからのスキャン動作に必要なパラメータを保存する。アプリケーションからの要求はRAM2002に保存される。スキャン動作管理8203はJob Manager1519とDIS7102からスキャンを行うのに必要な情報を取得する。スキャン動作管理8203はJob Manager1519から図18に示すジョブ番号、ドキュメント番号のテーブルデータ8301を受け取り、ジョブ番号、ドキュメント番号のテーブルデータ8301を元に、DIS7102より図19に示すスキャンパラメータ8302を受け取る。これによりアプリケーションから要求されているスキャン条件を元にスキャンを行う。

【0093】スキャン動作管理8203はDIS7102から取得したスキャンパラメータ8302をドキュメント番号順にスキャンシーケンス制御部8204に渡す。スキャンパラメータ8302を受け取ったスキャンシーケンス制御部8204はスキャン画像属性8308の内容に従ってデバイス1/F制御部8207をコントロールする。これにより図1の画像バス2008に接続されたデバイス1/F2020を動作させることにより、ケーブル2071を介してスキャン2070に制御コマンドを送ることによりスキャンが実行される。スキャンした画像はケーブル2071を介してデバイス1/F2020に渡り、さらにバス2008を介してRAM2002に格納される。

【0094】スキャンシーケンス制御部8204はスキャンが終了し、バス2008を介してRAM2002に画像が格納された時点で、スキャンパラメータ8302のスキャン画像圧縮形式8309の内容にしたがって、RAM2002に格納されているスキャン画像を圧縮するために、CODEC Manager1520に対して要求を出

す。要求を受け取ったCODEC Manager 1520はバス2008に接続されているコーデック2040、あるいはMMR-CODEC 1523内のソフトウェア圧縮モジュールを用いて、スキャンシーケンス制御部8204からのスキャン画像圧縮形式8309の指定で圧縮を行う。画像圧縮伸長部2043は圧縮された画像をバス2008を介してRAM2002に格納する。

【0095】スキャンシーケンス制御部8204はCODEC Manager 1520がスキャン画像圧縮形式8309で指定された形式でスキャン画像を圧縮し、RAM2002に格納した時点で、スキャンパラメータ8302の画像ファイルタイプ8307にしたがってRAM2002に格納されている圧縮されたスキャン画像をファイル化する。スキャンシーケンス制御部8204はファイルシステム8206に対して、スキャンパラメータ8302の画像ファイルタイプ8307で指定されたファイル形式でファイル化することを要求する。ファイルシステム8206はスキャンシーケンス制御部8204からの画像ファイルタイプ8307にしたがって、RAM2002に格納されている圧縮された画像をファイル化し、バス2008を介してHDD2004に転送することによりスキャンされた圧縮画像をファイル化する。スキャンシーケンス制御部8204はファイルシステム8206がHDD2004にファイル化された画像を格納した時点で、スキャナ2070上の一枚の原稿の処理が終了したとして、スキャン動作管理部8203にスキャン終了通知を送り返す。

【0096】この時点でスキャナ2070上にまだスキャンが行われていない原稿が存在し、Job Manager 1519からスキャン要求が存在する場合には再度、DIS7102に格納されているスキャンパラメータ8302を用いてスキャンシーケンス制御部にスキャン動作を要求する。スキャナ2070上にスキャンされていない原稿が存在しない場合、またはJob Manager 1519からのスキャン要求が存在しない場合には、スキャン動作が終了したものと見てJob Manager 1519に対してスキャン終了通知を発行する。

【0097】プリント動作

以下、プリント動作について詳細に説明する。デバイスI/F2020は内部にDPRAMを持ち、このDPRAMを介してプリンタへのパラメータ設定及びプリンタの状態読み出しと、プリントの制御コマンドのやりとりを行う。またこのボードはVideoコントローラを持ち、プリンタからエンジンインターフェースケーブル経由で与えられるVCLK (Video Clock) とHSYNCに合わせて、バス2008上に展開されているイメージデータをエンジンインターフェースケーブルを介してプリンタに送信する。この送信のタイミングを図で表すと図20のようになる。VCLKは常に出続け、HSYNCがプリンタの1ラインの開始に同期して与えられる。Video

コントローラは設定された画像幅(WIDTH)分のデータを、RAM2002から読み出して、Video信号としてエンジンインターフェースケーブルに出力する。これを指定ライン分(LINES)繰り返した後、IMAGE_END割り込みを発生する。

【0098】先に説明したとおり、CPU上のアプリケーションプログラムからControl APIにプリントジョブの指示が渡されると、Control APIはこれをコントローラレベルのJob Manager 1519にジョブとして渡す。さらにこのJob Manager 1519はジョブの設定をDIS7102に格納し、Print Manager 1526にジョブの開始を指示する。Print Manager 1526はジョブを受け付けるとDISからジョブ実行に必要な情報を読み出しEngine I/Fボード及び、DPRAMを介してプリンタに設定する。画像が圧縮されている場合はCODEC Managerに展開を依頼し、CODEC Managerは依頼に従ってプリンタ Managerから指示された展開方法(JPEG、MMRなど)により画像ファイルからビットマップ画像へ展開する。展開された画像はRAM2002に格納される。

【0099】デバイスI/F2020の設定項目を図21に、プリンタのDPRAMを介した設定項目及び制御コマンド、状態コマンドを図22に示す。ビットマップ画像の印刷についてレター(11"x8.5")サイズ2値画像の、2ページ1部プリント、プリンタが600dpiの性能を持つものとして、具体的に動作を説明する。

【0100】画像の展開終了後、Print Managerはこの画像の幅(この場合8.5"の側とする)の画像バイト数を算出する。
WIDTH=8.5×600÷8÷630(Bytes)
次にライン数を算出する。

LINES=11×600=6600(Lines)

【0101】これらの算出した値と、与えられた1ページ目の画像が格納されているRAM2002のSOURCEアドレスを図21に示したWIDTH、LINES、SOURCEに設定する。この時点でデバイスI/Fは画像出力の用意が完了しているが、プリンタからのHSYNC信号が来ていないため(VCLKは来ている)画像データを出していない。

【0102】次にPrint Managerは図22に示したDPRAMの所定のアドレス(Book No)に出力部数である1を書き込む。その後、1ページ目に対する出力用紙の給紙要求(FEED_REQ)を出し、プリンタからのIMAGE-REQを待つ。プリンタからIMAGE-REQが来たら、IMAGE_STARTを出す。これを受けてプリンタはHSYNCを出し始め、HSYNC待ちで、あったデバイスI/F2020は画像を出力する。プリンタは出力用紙の後端を検出すると、IMAGE_ENDを出力し、出力用紙が排出されるとSHEET_OUTを出力する。Print Managerは1ページ目のIMAGE_B

NDを受けて、2ページ目のWIDTH, LINES, SOURCEをEngine I/Fボードに設定し、FEED_REQを出して、IMAGE_REQを持つ。2ページ目のIMAGE_REQが来てからの動作は、1ページ目と同様である。

【0103】【コーデックの選択】以下に本発明の特徴であるソフトウェア（S/W）コーデック、ハードウェア（H/W）コーデックの選択について詳述する。まず、スキャン、プリント時のデータの流れタイミングの説明をする。スキャンの一例としてUniversal Send機能の場合を説明する。スキャンによってスキャンされた画像はまずデバイスI/F2020からバス2008を介して、RAM2002に書き込まれる。次にRAM2002から読み出されハードウェア、あるいはソフトウェアにより符号化された後、再度RAM2002に格納される。

【0104】ハードウェアによる符号化の場合バス2008を介してコーデック2040によって圧縮される。またソフトウェアによる圧縮の場合、RAM2002から読み出しCPU2001によって圧縮しRAM2002に格納される。RAM2002に格納された画像はTIFFファイルなどの汎用フォーマットに変換され、RAM2002、HDD2004に格納される。最後に、RAM2002、HDD2004から読み出されたファイルがバス2008、ネットワークI/F2010を介してネットワーク上の他の機器に転送される。

【0105】図23は、この一連の動作を行ったときのバス2008の使用状況をタイムチャートとして示した図である。図において、バス2008のデータ転送レートは平均2Mバイト/秒として考えた。データ量はスキャンがA4、600dpiを考えた4Mバイト、コーデック2040は圧縮率1/2であるとし（つまり入力4Mバイト、出力2Mバイトで合計）6Mバイト、LANは2Mバイトである。スキャン終了後圧縮、圧縮終了後LANで転送するため、各機能が単独で画像バスを使用する。

【0106】同様にプリント動作の一例としてRemote Copy Print機能を説明する。ネットワークI/F2010及びバス2008を介して送られてきた画像データ（TIFF化された圧縮画像）はまずRAM2002、HDD2004に格納される。次にTIFFなどの余分なヘッダーを除去した後、圧縮データがRAM2002に格納される。RAM2002から読み出したデータをハードウェア（バス2008→コーデック2040→バス2008）、あるいはソフトウェアにより復号化し、再度RAM2002に格納する。最後にビットマップ化された画像をRAM2002から読み出しバス2008、デバイスI/Fを介してプリンタ2095でプリント出力する。

【0107】ここで上記Universal Send機能におけるスキャン（原稿2枚）要求と、RemoteCopy Print機能によ

るプリント（1枚）要求が同時に発生した場合のタイミングを図24を用いて説明する。ここではあるジョブ、例えばスキャンジョブと並行してプリントジョブを行うような並列処理を対象としている。11001はスキャンジョブによる1枚目のスキャン、11002は同2枚目のスキャン、11003はプリントジョブによるプリントである。

【0108】1枚目のスキャン動作時、H/Wコーデックとプリント動作のLANが同時にバス2008を使用するため、2つの機能によるバスの競合が生じ処理時間が単独使用時の倍の時間かけて行われる。具体的にはH/Wコーデックは時刻2の時点で処理を開始し、時刻3では残り4Mバイト、時刻4では残り2Mバイトになるが、この時点でプリントジョブのLANへの転送（2Mバイト）が開始するため、ここからのデータ転送はH/Wコーデック、LANともに2Mバイトのデータ転送を時間単位2かけでおこなうことになる。

【0109】さらに時刻6では1枚目のスキャンのLANからの転送、2枚目のスキャンデータの転送、H/Wコーデックによるプリントデータのデコードが同時に行われることになり、3つの要求によるバスの競合が生じるため処理速度は単独使用時の1/3になる。

【0110】図25はこの場合のデータ転送の詳細を示す。LAN、スキャン、H/Wコーデックの各データ転送はLongWord（1LongWord=32ビット）単位で行い、順番にバスに転送される。

【0111】そこで本発明においてはスキャン、プリント、LANなどバス2008を共有する資源のバス使用状況を調査する。具体的にはH/Wコーデック2040に数100バイトから数1000バイト程度の疑似データをエンコードあるいはデコード（画像処理をしないスルーバスがあるならそれでもよい）、その処理時間が所定値以上ならバスが混雑しているためソフトウェアコーデックを使用し、そうでなければハードウェアコーデックを使用するように切り替える。

【0112】図26に示すようにCodec Manager1520はS/Wコーデック12001とH/Wコーデック12002のそれぞれに符号化（圧縮）/復号化（伸長）の指示を行う。ScanManager1524、あるいはPrint Manager1526から画像の圧縮/伸長依頼を受けたときCodec Manager1520はH/Wコーデック12002に疑似データのエンコード（あるいはデコード）を指示する。処理にかかった時間の情報はTimerLib12003に問い合わせることにより得られる。

【0113】次に、図27に示すフローチャートを用いてCodec Manager1520の動作を説明する。ステップS16001において、起動時、PCIコンフィグレーションの設定、及びコーデックチップのレジスタ初期化など、H/Wコーデックのイニシャライズを行う。ステップS16002においてScan Manager1524、Print

t Manager 1 5 2 6 から圧縮／伸長要求があるかどうかを判断する。要求がなければ終了し、要求がある場合、ステップ S 1 6 0 0 3 において Time Lib 1 2 0 0 3 から現在の時刻を取得する。

【0114】ステップ S 1 6 0 0 4 において RAM 2 0 0 2 に格納されている疑似データの圧縮（あるいは伸長）の開始を指示し、ステップ S 1 6 0 0 5 においてその完了の判断を行う。もし終了していなければ終了するまで待ち、もし終了していればステップ S 1 6 0 0 6 において再度 Time Lib 1 2 0 0 3 から現在の時刻を取得する。疑似データの圧縮（あるいは伸長）にかかった時間をステップ S 1 6 0 0 3、及びステップ S 1 6 0 0 6 で得られた 2 つの時刻の差によりステップ S 1 6 0 0 7 で求める。

【0115】ステップ S 1 6 0 0 8 の条件判断でステップ S 1 6 0 0 7 で求めた時間が所定値より小さければ、バスの負荷は小さく、H/W コーデックの方が速いと判断できるのでステップ S 1 6 0 0 9 で H/W コーデックを実行する指示を出す。ステップ S 1 6 0 1 0 で処理の終了を判断し、終了したらステップ S 1 6 0 0 2 へ戻る。またステップ S 1 6 0 0 8 でステップ S 1 6 0 0 7 で求めた時間が所定値と等しいか大きければ、バスの負荷は大きく S/W コーデックの方が速いと判断できるのでステップ S 1 6 0 1 1 で S/W コーデックを実行する指示を行う。ステップ S 1 6 0 1 2 で処理の終了を判断し、終了したらステップ S 1 6 0 0 2 へ戻る。

【0116】図 24 の例では (1) 1 枚目のスキャン (1 1 0 0 1) における H/W コーデック処理、(2) 1 枚目のプリント (1 1 0 0 3) の H/W コーデック処理、(3) 2 枚目のスキャン (1 1 0 0 2) の H/W コーデック処理前に図 27 に示した Codec Manager 1 5 2 0 によるコーデック選択処理が行われる。

【0117】図 28 に、図 24 と同じジョブスケジュールにおいて上述のコーデック選択処理を適用した場合の処理ダイアグラムを示す。ここでは、疑似データ 1 0 2 4 バイト、S/W コーデックと H/W コーデックとの選択を決定する所要値 (ステップ S 1 6 0 0 8 で使用) が 1 ミリ秒だと設定されているとする。また、図 23 と同様、バス 2 0 0 8 のデータ転送レートは平均 2 M バイト／秒として考えた。データ量はスキャンが A 4、6 0 0 d p i を考え 4 M バイト、画像圧縮 2 0 4 0 は圧縮率 1 / 2 であるとし (つまり入力 4 M バイト、出力 2 M バイト) 6 M バイト、LAN は 2 M バイトである。スキャン終了後圧縮、圧縮終了後 LAN で転送するため、各機能が単独で画像バスを使用するものとする。図 28 下にこれらの処理の進行状況とバス 2 0 0 8 の使用状況を時間単位毎に示す。

【0118】まず、1 枚目のスキャン処理 (1 7 0 0 1) において、疑似データを H/W コーデックに符号化させ、その処理が 7 0 0 マイクロ秒で処理が終わったと

すると、1 ミリ秒よりも短いため、H/W コーデックが選択される。次いで、1 枚目のプリント処理 (1 7 0 0 3) において疑似データをエンコードして 2 ミリ秒で処理が終わったとすると、S/W コーデックが選択される。また、2 枚目のスキャン処理 (1 7 0 0 2) において疑似データを H/W コーデックで符号化させ 6 0 0 マイクロ秒で処理が終わったとすると、H/W コーデックが選択される。このような選択処理の結果、図 28 は図 24 の例に対してスキャン、プリントジョブともに速く終了していることがわかる。

【0119】(第 2 の実施形態) 上述の第 1 の実施形態においては、画像バス 2 0 0 8 の占有率を元になぜのコーデックを用いるかを決定していたが、本実施形態においてはコーデックの使用状況によって使用するコーデックを選択することを特徴とする。なお、本実施形態の説明においては、MMRCodec 1 5 2 3 (図 12) が S/W コーデックを実装しており、また画像圧縮伸長部 2 0 4 3 (図 6) が MMR の H/W コーデックであって、それら 2 つのコーデックを処理に割り当てる場合を例にして説明する。また、本実施形態において、画像圧縮伸長部 2 0 4 3 は内部に複数の H/W コーデックを有する構成であっても良い。

【0120】図 29 は本実施形態における Coded Manager 1 5 2 0 と MMRCodec 1 5 2 3 内部のソフトウェアモジュール構成例を示すブロック図である。Coded Manager 1 5 2 0 の内部は、MMRCodec 1 5 2 3 に対して符号化／復号化の指示を行うコーデック動作指示部 1 5 2 0 1 と、コーデックを資源として管理しているリソース管理部 1 5 2 0 2 からなる。Codec Manager 1 5 2 0 は Scan Manager 1 5 2 4 (図 12) や Print Manager 1 5 2 6 の要求によって符号化／復号化を行うが、要求を受けたとき、まずリソース管理部 1 5 2 0 2 によってコーデック資源を確保する。そして、確保できたならば、MMRCodec 1 5 2 3 に処理を依頼する。

【0121】MMRCodec 1 5 2 3 内の MMR 動作指示部 1 5 2 3 1 は処理の指示を受けたとき、資源が H/W コーデック、S/W コーデックに関係なく、共通インタフェース 1 5 2 3 2 を介して、コーデックに処理を要求する。コーデックに対するインタフェースを共通化することで、H/W コーデックが S/W コーデックに変わったとき、またその逆の場合でも共通インタフェース 1 5 2 3 2 より下位層を変更すればよい。

【0122】共通インタフェース 1 5 2 3 2 は、処理の要求先が H/W コーデックか S/W コーデックかを識別して、処理の要求を H/W コーデック制御部 1 5 2 3 3 と S/W コーデック 1 5 2 3 4 に振り分ける。

【0123】H/W コーデック制御部 1 5 2 3 3 は H/W コーデックでの符号化、復号化を指示されたとき、画像圧縮伸長部 2 0 4 3 (図 6) に指示を行い、符号化、復号化を行う。また、S/W コーデック 1 5 2 3 4 は S

／Wコーデックでの符号化、復号化を指示されたとき、CPU2001によって符号化、復号化を行う。

【0124】（コーデック割り当て処理）次に、図30に示すフローチャートを用いて、リソース管理部15202のリソース要求受信時の動作を説明する。まず、ステップS101でコーデック動作指示部15201からのリソース要求を待つ。ステップS102でもし要求がなければ再びリソース要求待ちステップS101に戻る。もし要求があればステップS103でH／Wコーデックに空きがあるかどうかを調査する。リソース管理部15202はH／Wコーデック、S／Wコーデックの数と、使用しているものの数、また、どのコーデックが使用中であるかを管理している。

【0125】図31は、リソース管理部15202におけるコーデック管理の例を示す図である。リソース管理部15202はコーデックを図31示すようなテーブル形式で管理可能である。図において、コーデック種別領域310はH／W及びS／Wコーデックの種別を、使用中コーデック数領域311は、コーデックが複数ある場合に使用中であるコーデックの数を種別毎に記録する領域、番号領域312は各種別毎のコーデック番号を記憶する領域である。また、状況フラグ領域313には各コーデックが使用中である場合にはYes、空きである場合はNoがそれぞれ記憶される。

【0126】図31の例では、リソース管理部15202が各5つのH／Wコーデック及びS／Wコーデックを管理しており、H／Wコーデックは3つ、S／Wコーデックは2つがそれぞれ処理中である場合を示している。

【0127】図30に戻って、ステップS103において、H／Wコーデックに空きがある場合はリソース管理部15202におけるH／Wコーデックの使用コーデック数領域の値をインクリメントし（ステップS105）、割り当てるコーデックの状況フラグ領域の値をNoからYesに変えて、そのH／Wコーデックを割り当てる（ステップS106）。

【0128】一方、ステップS103において、H／Wコーデックに空きが無い場合はステップS104でS／Wコーデックの空きを調査する。S／Wコーデックに空きがある場合は、リソース管理部15202におけるS／Wコーデックの使用コーデック数領域の値をインクリメントし（ステップS107）、割り当てるコーデックの状況フラグ領域の値をNoからYesに変えて、そのS／Wコーデックを割り当てる（ステップS108）。

【0129】ステップS104において、S／Wコーデックにも空きがない場合には、リソース管理部15202が有するリソース待ちキューへリソース要求を書き込む（ステップS109）。

【0130】（コーデック割り当て解除処理）次に、図32に示すフローチャートを用いて、リソース管理部15202のリソース解放要求受信時の動作を説明する。

まず、ステップS201でコーデック動作指示部15201からのリソース解放要求を待つ。ステップS202でもし要求がなければ再びリソース要求待ちステップS201に戻る。もし要求があり、解放要求の対象がH／Wコーデックであれば（ステップS203）、リソース管理部15202におけるH／Wコーデックの使用コーデック数領域の値をデクリメントし、そのコーデックの状況フラグ領域の値をYesからNoに変える（ステップS204）。

【0131】一方、ステップS203において、解放要求の対象がS／Wコーデックであれば、リソース管理部15202におけるS／Wコーデックの使用コーデック数領域の値をデクリメントし、そのコーデックの状況フラグ領域の値をYesからNoに変える（ステップS205）。

【0132】使用中のコーデックを解放したら、ステップS206でリソース待ちキューのなかに待ち状態の要求があるかどうかをみる。もし、待ちがなければリソース解放要求待ちステップS201に戻る。

【0133】もし待ちの要求があれば、ステップS207でH／Wコーデックに空きがあるかどうかを調査する。H／Wコーデックに空きがある場合はリソース管理部15202におけるH／Wコーデックの使用コーデック数領域の値をインクリメントし（ステップS208）、割り当てるコーデックの状況フラグ領域の値をNoからYesに変えて、そのH／Wコーデックを割り当てる（ステップS210）。

【0134】一方、ステップS207において、H／Wコーデックに空きがない場合には、リソース管理部15202におけるS／Wコーデックの使用コーデック数領域の値をインクリメントし（ステップS209）、割り当てるコーデックの状況フラグ領域の値をNoからYesに変えて、そのS／Wコーデックを割り当てる（ステップS211）。

【0135】（コーデック動作指示部の動作）次に、図33に示すフローチャートを用いて、コーデック動作指示部15201の動作を説明する。ステップS301においてScan Manager1524あるいは、Print Manager1526からの符号化、復号化要求が、Job Manager1519を介して受信されるのを待つ。もし要求の受信がなければ（ステップS302）、要求待ちステップS301に戻る。要求がある場合、リソース管理部15202に対して、コーデックを要求する（ステップS303）。

【0136】コーデックが獲得できない場合は待ち続け（ステップS304）、コーデックが獲得できたときにはScan Manager1524あるいは、Print Manager1526からの要求に応じてMMRCodec1523に対してMMRの圧縮／伸長を要求する（ステップS305）。その後、圧縮／伸長処理が完了するまで待ち（ステップS3

06)、処理が完了したらステップS307でリソースを解放するためリソース管理部15202に対して、コーデックの解放要求する。その処理が完了するまで待ち(ステップS308)、完了したら、ステップS301で再度符号化/復号化要求を持つ。

【0137】

【他の実施形態】 上述の実施形態においては符号化/復号化処理を行うコーデックをソフトウェア、ハードウェア両方で実現した場合のみを説明したが、変倍や回転など一般的な画像処理をハードウェア及びソフトウェアで実行可能な場合であっても本発明を適用することができる。

【0138】 また、上述の実施形態においては画像データを取り扱う装置に本発明のデータ処理装置を適用した例を用いて説明したが、処理対象のデータは画像データに限定されるのではなく、音声、文書、プログラムなど、任意のデータを取り扱う装置に適用できることはいうまでもない。

【0139】 また、上述の実施形態においては、画像バス2008の占有率を元にソフトウェアコーデックの使用を決定していたが、ソフトウェアコーデックとして動作するCPU2001が他の処理によって高い負荷状態にあった場合には、ハードウェアコーデックを使用した方が結果として処理が早いことも考えられる。そのため、CPU2001の負荷状態と画像バス2008の占有率とを組み合わせるソフトウェアコーデックの使用を決定するように構成しても良い。

【0140】 なお、本発明は、複数の機器(例えばホストコンピュータ、インタフェイス機器、リーダ、プリンタなど)から構成されるシステムに適用しても、一つの機器からなる装置(例えば、複写機、ファクシミリ装置など)に適用してもよい。

【0141】 また、本発明の目的は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体(または記録媒体)を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ(またはCPUやMPU)が記憶媒体に格納されたプログラムコードを読み出し実行することによっても、達成されることは言うまでもない。この場合、記憶媒体から読み出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。また、コンピュータが読み出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているオペレーティングシステム(OS)などが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0142】 さらに、記憶媒体から読み出されたプロ

ラムコードが、コンピュータに挿入された機能拡張カードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張カードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0143】 本発明を上記記憶媒体に適用する場合、その記憶媒体には、先に説明した(図27、図30、図32及び図33のいずれか1つ以上に示す)フローチャートに対応するプログラムコードが格納されることになる。

【0144】

【発明の効果】 以上説明したように本発明によれば、所定のデータ処理をソフトウェアで行うソフトウェア処理部と、共通バスに接続された所定のデータ処理をハードウェアで行うハードウェア処理部とを設けるとともに、処理の内容や処理部の使用状況などによって使用する処理部を選択することにより、効率の良い処理部の選択が可能になり、処理全体が完了するまでの時間を最適化できるという効果を得る。

【図面の簡単な説明】

【図1】 本発明を適用した複合機の全体構成を示すブロック図である。

【図2】 本発明を適用した複合機の正面外観を示す図である。

【図3】 操作部外観を示す図である。

【図4】 スキャナ画像処理部のブロック図である。

【図5】 プリント画像処理部のブロック図である。

【図6】 コーデックのブロック図である。

【図7】 画像回転部のブロック図である。

【図8】 画像回転処理の説明図である。

【図9】 画像回転処理の説明図である。

【図10】 デバイスI/F部のブロック図である。

【図11】 本発明を適用した装置を接続したネットワークシステムの構成例を示す図である。

【図12】 本発明を適用した複合機のソフトウェアの全体構成図である。

【図13】 配信用組み込みアプリケーションを説明するブロック図である。

【図14】 操作部の具体的表示例を示す図である。

【図15】 DISとJob Manager、Print Manager、Scan Managerとの接続関係を示す図である。

【図16】 DIS内部のデータベース、及びカウンタを示す図である。

【図17】 スキャンにおけるソフトウェア制御のブロック図である。

【図18】 スキャンにおけるパラメータテーブルの概略図である。

【図19】 スキャンにおけるパラメータテーブルの概略

図である。

【図 20】プリント時のデータの転送タイミングを示す波形図である。

【図 21】Engine I / F ボード内のプリントパラメータレジスタを示す図である。

【図 22】プリンタとEngine I / F ボードとの通信コマンドを示す図である。

【図 23】1 ページスキャン時のタイムチャートである。

【図 24】複数のジョブが同時発生した場合のタイムチャートである。

【図 25】3 つの処理が重複した場合のデータ転送順序を示す図である。

【図 26】コーデックマネージャ周辺のソフトウェア構成を示す図である。

【図 27】第 1 の実施形態におけるコーデックの動作を示すフローチャートである。

【図 28】図 27 において第 1 の実施形態によるコーデック切り替えを行った際のタイムチャートである。

【図 29】第 2 の実施形態におけるコーデックマネージャ及びMMR コーデックのソフトウェア構成を示す図である。

【図 30】第 2 の実施形態におけるコーデック割り当て

処理を説明するフローチャートである。

【図 31】リソース管理部 15202 における管理テーブルの例を示す図である。

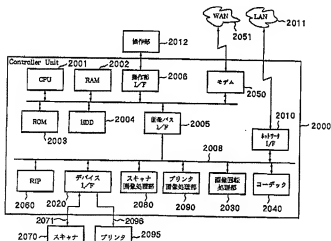
【図 32】第 2 の実施形態におけるリソース解放処理を説明するフローチャートである。

【図 33】第 2 の実施形態におけるコーデック動作指示部の動作を説明するフローチャートである。

【符号の説明】

- 2001 CPU
- 2002 RAM
- 2003 ROM
- 2004 HDD
- 2005 画像バスインタフェース
- 2006 操作部インタフェース
- 2007 システムバス
- 2008 画像バス
- 2010 ネットワークインタフェース
- 2020 デバイスインタフェース
- 2040 コーデック
- 2050 モデム
- 2060 ラスタイメージプロセッサ
- 2070 スキャナ
- 2095 プリンタ

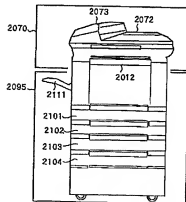
【図 1】



【図 18】

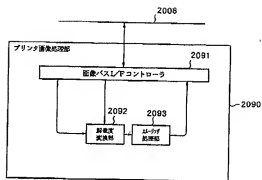


【図 21】

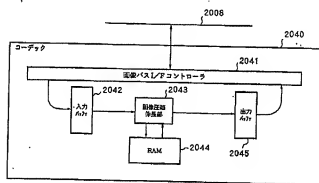


【図 2】

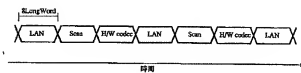
【図 5】



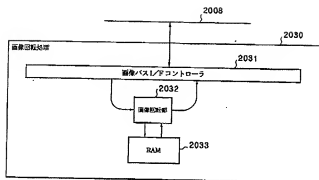
【図 6】



【図 25】



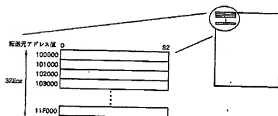
【図 7】



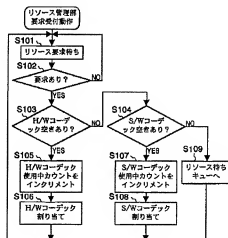
【図 31】

種別	使用中 コーデック数	番号	使用中
H/W コーデック	3	1	YES
		2	NO
		3	YES
		4	YES
		5	NO
S/W コーデック	2	1	NO
		2	YES
		3	NO
		4	NO
		5	YES

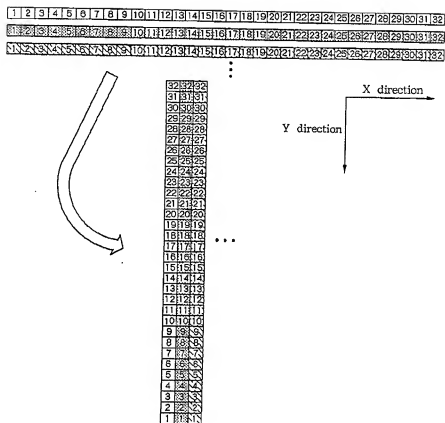
【図 8】



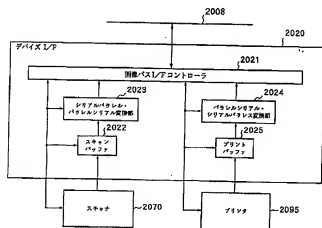
【図 30】



【図 9】



【図 10】

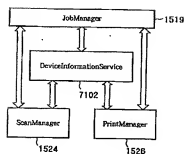


```

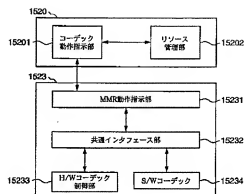
graph TD
    Start([リソース管理  
解放実行動作]) --> S201[S201  
リソース解放要求待ち]
    S201 --> S202[S202  
要求あり?]
    S202 -- NO --> S205[S205  
S/Wコードが使用中  
カウントをデクリメント]
    S202 -- YES --> S203{S203  
H/W  
コード?}
    S203 -- NO --> S205
    S203 -- YES --> S204[S204  
H/Wコードが使用中  
カウントをデクリメント]
    S204 --> S206{S206  
キューに  
待ちあり?}
    S206 -- NO --> S205
    S206 -- YES --> S207{S207  
H/Wコード  
タイプあり?}
    S207 -- YES --> S208[S208  
H/Wコードが使用中  
カウントをインクリメント]
    S207 -- NO --> S210[S210  
S/Wコードが使用中  
カウントをインクリメント]
    S208 --> S209[S209  
H/Wコードが解放して]
    S210 --> S211[S211  
S/Wコードが解放して]
    S209 --> End([解放完了])
    S211 --> End
  
```

FIG. 10 is a flowchart illustrating the H/W code check process. The process begins with a start node (リソース管理 解放実行動作). It proceeds to step S201 (リソース解放要求待ち). In S202, it checks if there is a request (要求あり?). If NO, it proceeds to S205 (S/Wコードが使用中 カウントをデクリメント). If YES, it proceeds to S203 (H/Wコード?). In S203, if NO, it proceeds to S205. If YES, it proceeds to S204 (H/Wコードが使用中 カウントをデクリメント). In S204, it proceeds to S206 (キューに待ちあり?). In S206, if NO, it proceeds to S205. If YES, it proceeds to S207 (H/Wコードタイプあり?). In S207, if YES, it proceeds to S208 (H/Wコードが使用中 カウントをインクリメント). In S207, if NO, it proceeds to S210 (S/Wコードが使用中 カウントをインクリメント). In S208, it proceeds to S209 (H/Wコードが解放して). In S210, it proceeds to S211 (S/Wコードが解放して). Both S209 and S211 lead to the end node (解放完了).

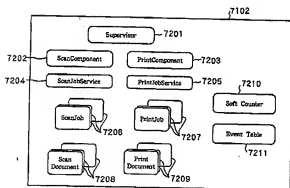
【図15】



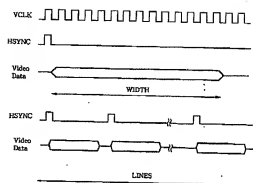
【図29】



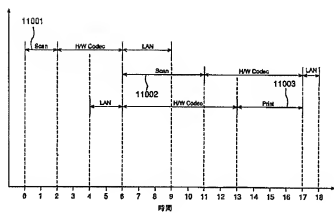
【図16】



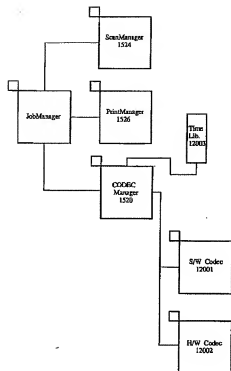
【図20】



【図24】

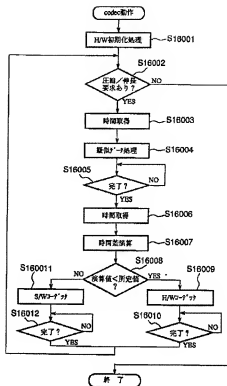


【図 26】



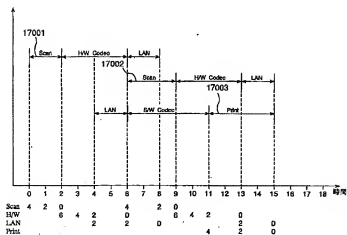
00 (/)

【図 27】



00 (/)

【図 28】



【図 3 3】

